**US Army Corps
of Engineers**
Construction Engineering
Research Laboratory

AD-A239 107

Organizational Response to the Introduction of New Computer Software Technology

by
Janet H. Spoonamore
Ben J. Sliwinski
Charles E. Kozoll

Employee productivity is a management issue of fundamental importance for the U.S. Army. Personal computer (PC) technology offers a way to dramatically boost a worker's productivity, but frequently it appears that users do not exploit this technology to its full potential. The organizational culture of the workplace may influence how effectively employees adopt new productivity technology.

This report summarizes basic research on the process by which employees adopt new computer software technology. Detailed interviews were conducted with a representative sample of research and administrative personnel at the U.S. Army Construction Engineering Research Laboratory (USACERL) to discover how the laboratory's organizational culture influences the adoption of new software technology. The subjects were questioned about their experiences learning and using word processors, computer-aided drafting and design, and spreadsheet programs.

It was found that the ability of USACERL employees to use a new software feature generally depends on the extent to which it can produce immediate, tangible results on the job. Also, users tend to ignore valuable new technologies until a specific job demand creates the need to learn. Subjects preferred learning informally from trusted colleagues over attending workshops or using prepackaged tutorials.

Recommendations for promoting technology adoption are included.

DTIC
ELECTE
AUG 0 5 1991
S
B
D

91-06839

91 8 05 012

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official indorsement or approval of the use of such commercial products. The findings of this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE July 1991	3. REPORT TYPE AND DATES COVERED Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Organizational Response to the Introduction of New Computer Software Technology	PE 4A161102 PR AT23

6. AUTHOR(S)
Janet H. Spoonamore, Ben J. Sliwinski, and Charles E. Kozoll

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
U.S. Army Construction Engineering Research Laboratory (USACERL) P. O. Box 9005 Champaign, IL 61826-9005	CERL TR P-91/39

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQUSACE 20 Massachusetts Avenue, NW Washington, DC 20314-1000	

11. SUPPLEMENTARY NOTES
Copies are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for public release; distribution is unlimited.	

13. ABSTRACT (Maximum 200 words)

Employee productivity is a management issue of fundamental importance for the U.S. Army. Personal computer (PC) technology offers a way to dramatically boost a worker's productivity, but frequently it appears that users do not exploit this technology to its full potential. The organizational culture of the workplace may influence how effectively employees adopt new productivity technology.

This report summarizes basic research on the process by which employees adopt new computer software technology. Detailed interviews were conducted with a representative sample of research and administrative personnel at the U.S. Army Construction Engineering Research Laboratory (USACERL) to discover how the laboratory's organizational culture influences the adoption of new software technology. The subjects were questioned about their experiences learning and using word processors, computer-aided drafting and design, and spreadsheet programs.

It was found that the ability of USACERL employees to use a new software feature generally depends on the extent to which it can produce immediate, tangible results on the job. Also, users tend to ignore valuable new technologies until a specific job demand creates the need to learn. Subjects preferred learning informally from trusted colleagues over attending workshops or using prepackaged tutorials.

Recommendations for promoting technology adoption are included.

14. SUBJECT TERMS		15. NUMBER OF PAGES 32
productivity new technology	computer technology government employees	
		16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT SAR

TRADEMARK NOTICE

AutoCAD is a registered trademark of AutoDesk, Inc.

Freelance is a registered trademark of Lotus Development Corporation.

Lotus 1-2-3 is a U.S. registered trademark of Lotus Development Corporation.

Microsoft Excel is a registered trademark of Microsoft Corporation.

Microsoft Word is a registered trademark of Microsoft Corporation.

ProComm is a registered trademark of Datastorm Technologies, Inc.

UNIX is a registered trademark of AT&T Information Systems.

Ventura Publisher is a registered trademark of Ventura Software, Inc.

Wingz is a trademark of Informix Software.

WordPerfect is a registered trademark of WordPerfect Corporation.

Other product names mentioned in this report are trademarks or registered trademarks of their respective holders.

FOREWORD

This study was conducted for Headquarters, U.S. Army Corps of Engineers (HQUSACE) under Project 4A161102AT23, "Basic Research in Military Construction"; Work Unit "Organizational Response to the Introduction of New Technology."

This research was conducted by Facilities Systems (FS) Division of the U.S. Army Construction Engineering Research Laboratory (USACERL). The principal investigator was Janet H. Spoonamore. The work was performed with the assistance of Ben J. Sliwinski of Research Associates, Urbana, IL, and Charles E. Kozoll of the Office of Continuing Education, University of Illinois, Urbana. Dr. Michael J. O'Connor is Chief of USACERL-FS. The USACERL technical editor was Gordon L. Cohen, Information Management Office.

COL Everett R. Thomas is Commander and Director of USACERL, and Dr. L.R. Shaffer is Technical Director.

DTIC
COPY
INSPECTED
6

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	

By_____
Distribution/

Availability Codes	
Dist	Avail and/or Special
A-1	

3

CONTENTS

ORGANIZATIONAL RESPONSE TO THE INTRODUCTION OF NEW COMPUTER SOFTWARE TECHNOLOGY

1 INTRODUCTION

Background

The U.S. Army Corps of Engineers (USACE), in performing its many civilian and military missions, relies to a great extent on computers and computer software. The effectiveness of the technology transfer process that occurs as new software is introduced within the Corps has a significant effect on how well the Corps performs its mission. As in many large organizations, the introduction of new software does not always proceed smoothly and, in some cases, the full benefits of the technology are not realized. As microcomputers have become widely used within the Corps, the importance of software technology transfer has been amplified. Consequently, the Corps is seeking to identify the factors that govern software technology transfer. Both technical and support personnel vary in their familiarity with, and reliance on, computer technology. Apart from anecdotal information, little is known about the process of becoming computer literate. More research must be conducted on the process by which both technical and support staff improve their effectiveness as employees through the use of computer* systems. Of particular interest are the characteristics of the software, the potential user, and the process of introducing the software to the user in ways that lead to rapid and effective adoption of the technology.

Objective

The objective of this research was to identify problems in internal technology transfer of office productivity software at a USACE research laboratory and recommend ways to improve the process.

Approach

The approach taken in this work effort was to develop case studies of users' experiences with three types of software widely used at the U.S. Army Construction Engineering Research Laboratory (USACERL): word processors, computer-aided drafting and design (CADD) software, and spreadsheet programs.

The case study methodology employed in this basic research differs from conventional quantitative methods of surveying the attitudes of a population. It was used because generalization was not the project's primary objective. The case study methodology is useful for developing a detailed description and understanding of a social system at a specific time and place. The primary methods of data collection used in case studies are interviewing, observing, and gathering unobtrusive data.[1]

---

* In this report, the term "computer" refers to desktop personal computers unless otherwise specified.
[1] Robert T. Etter, *Integrating Microcomputers in the Workplace*, unpublished doctoral dissertation (University of Illinois, 1989).

In the present effort, data were collected through taped interviews of USACERL personnel.[*] Interviews were conducted with five people who primarily use word processing software, five who work mainly with CADD software, and two who often use spreadsheet software. The spreadsheet user interviews represent an extension of the study that was prompted by observations made during the other interviews. Since the methodology of this user survey was qualitative rather than quantitative, extending the study in this manner was considered consistent with, and supportive of, the study's objective. The questionnaire used to guide the interviews is included in Appendix A.

The interviews were not conducted *en masse* as is usually the case in qualitative opinion surveys (e.g., marketing-style focus groups), even though the number of participants was right for such a group. The interviews were conducted individually as kind of a 'serial' focus group. This avoided one pitfall of focus groups: the peer pressure or "group think" that frequently skews the input of participants.[2]

The qualitative case study method employed here was determined to be most appropriate for the present objective. The focus group approach might be more appropriate for followup research—developing insight into why typical users like or dislike a particular word processing package, for example. The typical quantitative approach, on the other hand, would best be used to determine how many employees prefer one package over all others.

Scope

The authors imply no quantitative statistical significance for the findings of this study because quantitative methods were not attempted. The methodology employed here is related to qualitative techniques used in marketing research. This research was most concerned with the open-ended questions of "how" and "why" regarding software technology adoption at USACERL rather than the quantitative explanation of variations that is the focus of much opinion research. An important hypothesis is posed—that computer technologies have *not* been integrated into organizations to their fullest potential. A qualitative approach to this complex issue is central to identifying specific technology adoption barriers which, in the future, may be studied quantitatively. In terms of approach and results sought, this study has more in common with a focus group than with an opinion survey such as the Gallup Poll.

Applicability of This Research

Other institutions may consider the research and management issues discussed in this report to be a point of departure for an investigation of their own organizational response to the introduction of new computer technology. It must be emphasized, however, that the data and organizational characteristics reported here are specific to USACERL. No claim is made that the findings and recommendations apply universally to other similar settings. The requirements for validity dictate that studies of this type must be designed by professionals familiar with the organization and qualified to conduct qualitative opinion research.

---

[*] Although the subject matter of the interviews was not inherently controversial, interviewees were guaranteed anonymity with respect to the published report of the survey's findings. The researchers believed that subject anonymity would encourage the fullest and most candid responses from the interviewees, especially in subject areas pertaining to relevant laboratory management policies and practices. The survey included both male and female subjects, so the researchers used gender neutral language whenever practical to help preserve complete anonymity. In some passages, however, such language tended to bog down the narrative or produce bad grammar. In those cases the masculine pronouns "he," "him," or "his" were used for clarity while preserving anonymity.

[2] Easy Klein, "What You Can—and Can't—Learn From Focus Groups,' *D&B Reports* (Dun & Bradstreet, July/August 1989).

2   DESCRIPTION OF THE ORGANIZATION

USACERL, located in Champaign, Illinois, is one of four USACE research laboratories. The primary mission of USACERL is to provide research support to the Corps' mission of designing, building, and maintaining the infrastructure of military facilities. Because the Corps has additional missions and provides support to agencies other than the Army, the research programs at USACERL are diverse.

The laboratory is jointly administered by a military Commander and Director and a civilian Technical Director. The military position is rotational—a new Commander and Director is designated every 2 to 3 years—but the Technical Director position is permanent.

The research at USACERL is conducted within its four technical divisions: Facility Systems, Environmental, Energy and Utility Systems, and Engineering and Materials. Each is directed by a division chief supported by team leaders. There are five to eight teams per division each with about 20 members. Team leaders are responsible for the technical quality of the team's work. The technical staff includes principal investigators (who are responsible for specific projects) and the researchers and research assistants who support them.

In addition to the technical staff, each division has a dedicated support staff that includes a division secretary, administrative assistants, and clerk-typists. Technical teams may also have one or more administrative assistants and clerk-typists. These support personnel work as an integral part of the overall research effort at USACERL. The organization also includes a number of offices that provide administrative support (e.g., contract management, information management, resource management). Each office relies on personal computers (PCs) for a substantial amount of its productivity.

Some commercial PC applications (e.g., word processing) are used almost universally throughout the laboratory. Others (e.g., computer-aided drafting and design) are used almost exclusively by researchers. Appendix B discusses how three types of off-the-shelf software could reasonably be expected to promote employee productivity at USACERL.

Communications with clients about technology transfer and identification of research needs are handled by principal investigators, team leaders, division chiefs, and executive office personnel. Communications skills are well developed among the USACERL technical staff and throughout the laboratory.

Microcomputer technology is widespread and heavily used at USACERL. Every division chief, team leader, principal investigator, and researcher has a microcomputer, as do the majority of the research assistants. Furthermore, computer usage and acceptance by support personnel is equally high: nearly all workers in these categories also have their own computer or access to one. Most microcomputers in the lab are IBM-compatible, but about 15 percent are Apple Macintoshes. Most of the IBM-compatible computers use DOS*, but about 20 percent use Unix. In addition to the microcomputers the lab has a number of engineering workstations and minicomputers.

Electronic mail and a local area network (LAN) linking USACERL's minicomputers have been in service for nearly 10 years, and microcomputer-based LANs are established and growing. Because of the availability of computer resources, the laboratory's research mission, and the quality of personnel, computer literacy is the norm throughout the organization. However, as this report will show, both literacy and software familiarity vary significantly among both technical and support personnel. In some cases this may be expected, but in others it may indicate that some employees are not encouraged or highly motivated to increase their computer knowledge, skills, and productivity.

---

*DOS is Disk Operating System, the widely used operating system for IBM-compatible microcomputers.

3 SUMMARY OF INTERVIEWS

The taped interviews consisted of a section of background questions, a section pertaining to the user's introduction to work-related software, and a section exploring the user's experiences with that software (see Appendix A). Some overlap of subject matter among the sections was included by design. Objectives of the interview were to discuss user's perceptions of the software's utility and ease of use, and the methods used to learn the software. To maximize the amount of insight gained in this research, the interviews were not rigidly structured; supplemental questions were frequently asked to explore topics of opportunity.

This chapter summarizes the findings of the taped interviews. Comments by the respondents are provided to support the conclusions and recommendations made in the chapters that follow, but readers not requiring this level of detail may proceed directly to Chapter 4.

Background Questions

As previously stated, 12 interviews were conducted between October 10 and November 9, 1989. Of the users interviewed, two were team leaders, two were support personnel, and the remaining eight included principal investigators, full-time researchers, and part-time student research assistants. All technical personnel had bachelor of science degrees, and most had an advanced degree or were working toward one at the time of the interview. Their educational backgrounds included engineering, architecture, computer science, business administration, and finance. The two members of the support staff had completed high school; one was working toward an associate degree at a junior college. Ages of those interviewed ranged from about 20 to 50 years old.

The degree of continuing education that interviewees pursued through short courses varied, being minimal for those currently working on advanced degrees and increasing somewhat for those whose formal education was completed. The number of software training courses being taken was approximately one per year for those who took any. Most technical personnel did not consider training courses an effective way to learn software, except in the case of two AutoCAD users who learned that program as part of their formal education.

Conversely, one member of the support staff indicated a strong interest in training courses, and said that interaction among students in a classroom setting was helpful: they helped each other learn.

All technical personnel indicated a heavy reliance on computers in performing their daily work, with the majority claiming daily usage of more than 6 hours. When asked what type of software they primarily relied on, most users first indicated a word processor—WordPerfect or Microsoft Word. Other packages mentioned as useful included AutoCAD, BLAST,* Microsoft Excel, Lotus 1-2-3, Ventura Publisher (a page layout program), Freelance (a sophisticated drawing program), ProComm (a communications package), and USACERL's electronic mail program; several made unsolicited positive comments about Ventura and Procomm. All users surveyed exhibited a moderate to high level of computer literacy, and none exhibited any computer phobia, including several who had only recently begun using computers.

Most users said their preferred ways to learn software in general included trial and error, study of documentation, and assistance from peers and coworkers. Training sessions were the least favorite method. A few exceptions among users had spent time studying the documentation before trying software, but most tended to "jump in" and try it. Several said certain software was "intuitive," or that ideally

---

*Building Loads Analysis and Systems Thermodynamics, an energy analysis software package developed by USACERL.

software *should* be intuitive. In response to follow-up questioning, most defined "intuitive," as a quality of software logic that complemented their own. Instead of struggling to determine how to use a feature, its use was either apparent or easily accessible from a help menu, template, or brief explanation in the documentation.

A theme that emerged both in this section of the questionnaire and in the following section on user introduction of the software was the concept of a two-stage learning process. The first stage involved the process of trial and error with limited reference to the software documentation. These techniques would lift the user to a plateau where the minimum skills needed to perform simple (but useful) tasks with the software have been achieved. For example, a respondent using a new word processor would prepare and print a letter or memo. New users of AutoCAD established a comfort level by developing simple designs. The second stage involved moving from this plateau to a level of moderate fluency through interaction with peers and coworkers, who provide context-sensitive help. This would occur when a new user encountered a problem using a specific feature, or wondered if the software could be used to complete other work. Within the group surveyed, the new user was more likely to ask peers for help than to review the documentation for an answer or to study a new feature; only those who were comfortable with the index feature of the documentation chose to work on their own.

In many cases, users were able to identify peers and coworkers with specific areas of expertise on a given software package, and could identify their own particular areas of expertise. Informal help networks have developed within teams, particularly among groups that naturally work together. Both specific assistance and general discussion of a program's value and limitations occur regularly among users at USACERL.

When asked how they improved their familiarity with computer hardware and software, several indicated that they regularly referred to journals and magazines. These publications were used chiefly as a source of information about the overall value of different packages by personnel looking for software to complement packages they already used. The articles do not appear to be used very frequently as tools for increasing the user's familiarity with a program already used. One user, however, relied heavily on journals; he also owned his own computer and an extensive software library. The rest of the group interviewed did not tend to buy software for experimentation. Limited time, funds, and perhaps interest are possible reasons.

A theme that emerged was the importance of a trusted peer in providing input on the qualities of new software. Several of those interviewed said that they sought out people who they thought were "in the know," either at USACERL or elsewhere. Some also relied on certain writers who they felt shared their interests and were particularly knowledgeable. Vendors were also mentioned as a source of information on new hardware and software, but their views were treated cautiously: those who discussed this potential source of assistance expressed awareness of vendors' interest in selling products over assisting customers with a problem. Only one interviewee mentioned training courses as a source of information.

For the most part, everyone interviewed felt that they were supported by their supervisor in gaining additional hardware and software expertise. Some respondents, however, said supervisors *do* question the need for adding hardware features or new software. This implied a concern by managers about the ability of users to distinguish between needs versus luxuries.

Questions About User Introduction to Software

The second section of the interview focused on the user's introduction to a specific work-related software package. AutoCAD was specifically evaluated for reasons already given, but several word processors and spreadsheets were considered. The word processing programs discussed were WordPerfect, Microsoft Word, and Lotus Manuscript; the spreadsheets were Lotus 1-2-3 and Microsoft Excel.

The process employed to become familiar with a specific software package varied according to the user's previous experience, learning style, and the kind of expertise needed immediately. In the case of CADD software, the users interviewed had little choice since they were hired to work on projects for which USACERL had already selected AutoCAD. In the case of users of word processing and spreadsheet software, there was more room for user choice, even though WordPerfect is the stated laboratory standard; it is apparent that the WordPerfect standard is not strictly enforced if the user's alternative word processor can convert files to WordPerfect format, or its files can be accurately read by WordPerfect.

Most users first became familiar with the features of their word processor by seeing examples of work it produced and/or from information provided by respected peers and coworkers. In general, before users even tried the software they were convinced that the software provided a high degree of actual or potential utility. When convinced of the software's benefits, the users interviewed were prepared to deal with the frustrations associated with mastering the basics; the promise of software benefits provided this group a powerful incentive to learn. Users of both word processing and AutoCAD emphasized this point. In one case, a user of WordPerfect claimed that he was originally "frustrated to tears" and that "the WordPerfect tutorial was terrible." Yet, this user, motivated by a strong belief that the software had a lot to offer, persevered, became skilled with WordPerfect, and now highly recommends the program. Some AutoCAD users reported a similar experience; several indicated initial difficulties, but ultimately intended to master the program's features. The spreadsheet users also reported the same experience.

An important variation to this approach, however, was discovered among the more experienced users who were not content to struggle with a package that did not produce immediate results. They instead chose alternative word processing packages (i.e., Microsoft Word or Lotus Manuscript), which they had previously reviewed and found easier to use. These advanced users were more willing to make the effort to integrate their preferred software with programs used around the lab than to work with software that they find cumbersome, illogical, and of limited usefulness.

Among those who do word processing regularly, two dimensions of the software are important. One is its usefulness in everyday basic tasks; this involves how well the program blends with the user's established work style to make the drafting of routine documents on the computer easy. The other involves more advanced tasks the user must accomplish—formatting scientific equations, for example; the selection of software will, to some extent, be governed by its ability to handle certain frequently performed advanced tasks.

In general, the interviewees said self-paced learning was the most effective way to become familiar, comfortable, and reasonably proficient with work-related software, whether a word processor, AutoCAD, or a spreadsheet.

All of the word processor users interviewed began to master the software without formal training. Only one interviewee, a member of the support staff, expressed a desire for formal training.

The concept of a two-step learning process surfaced again in several interviews. Also, the term "intuitive" was again cited as a desirable software quality. Some users indicated difficulty not in learning their current word processor, but rather in *unlearning* aspects of their previous word processing package. Most users said they were not given a lot of time to learn a new word processor before regularly relying

10

on it; only a limited amount of time to experiment was available. In most cases, enough progress was made to complete the tasks immediately required. This experience was not the case, however, with the support staff member who desired formal training.

The experiences of AutoCAD users as they became familiar with AutoCAD were different from those of the word processing and spreadsheet users. In most cases, AutoCAD users learned to use the software as part of their formal education. Those who did not receive formal instruction received intensive one-on-one training whether before working at USACERL or after arriving. Additionally, AutoCAD users were strongly motivated to learn the software by their perception that it would be necessary for career advancement.

AutoCAD users also were motivated to some extent by exposure to the capabilities of other CADD packages, so they had some idea of the comparative utility of AutoCAD. Some users said they were impressed with AutoCAD's improved quality of drawings, its ability to perform three-dimensional perspectives and "fly arounds," and its elimination of repetitive work. Because all the AutoCAD users who were interviewed were hired onto projects already using AutoCAD, peer influence had little effect on determining which CADD package they would use. Similarly, those who had formal training were influenced to take courses offering AutoCAD, only to the extent that their peers had positive comments about the courses. Alternative courses teaching other CADD packages were not available to anyone interviewed. Several CADD users did indicate that decisions about their choice of add-on software modules or ways of improving AutoCAD technique were influenced by knowledgeable peers.

Initial difficulties cited by several AutoCAD users dealt with having to learn things the hard way. Often they would spend considerable time trying to accomplish a drawing function only to find later that one simple command could produce the same result. One user who had formal training said the order of teaching certain features should be changed, because it seemed as if the most useful commands were not taught until the end of the course. As users became more familiar with the program, they were impressed with its usefulness. One user said, "chances are, if you want to do something with that program, there is a way to do it."

An interesting aspect of two AutoCAD-related interviews was that the users had no previous computer experience before learning AutoCAD; these interviewees were learning AutoCAD and DOS at the same time. Both of them said that they learned AutoCAD faster than they learned DOS. One user said that most of their initial difficulties came from a lack of general computer knowledge, AutoCAD, they said, "was not totally intuitive, but pretty close."

The two users of spreadsheets included in the study were interviewed primarily to explore the concept of "intuitive software" that was emerging from the other interviews. The researchers felt that typing and drafting served as a kind of built-in intuitive basis for learning word processing or CADD programs respectively; spreadsheets, however, seemed not to have any such counterpart activity that would have entered significantly into a user's earlier experience. Therefore it was concluded that a user's reaction to learning spreadsheet software could offer insight into intuitiveness (or the lack of it) in software.

One of the spreadsheet users interviewed was a member of the technical staff and the other was a member of the support staff. The technical staff member was first introduced to Lotus 1-2-3 as part of a special project in civil engineering in which a professor suggested that a spreadsheet could be used effectively to analyze certain data. The support staff member learned of Lotus 1-2-3 through conversations with fellow students at a junior college; this interviewee's motivation for learning the software was stimulated by positive comments by peers as well as his own observation that experience with Lotus 1-2-3 was becoming a prerequisite for many employment opportunities.

11

Both spreadsheet users learned to use the software through trial and error, reference to documentation, and consulting with coworkers. The respondent from the technical staff indicated that he did not have many problems with the spreadsheet software, but nevertheless said that Lotus was not particularly easy to learn. "Lotus did not seem intuitive a lot of times, but I'm the kind of person that perseveres," this interviewee said. When questioned further about areas that did not seem intuitive, this user mentioned writing equations in cells and producing graphs—particularly manipulating the format of the graph.

The support staff member's experience with Lotus 1-2-3 at the time of the interview was limited to having developed a few spreadsheets after doing the Lotus tutorial. He reported no problems using the program at this level.

Questions About Working Experience With Software

The final portion of the interview examined users' attitudes about their work-related software after becoming familiar enough with it to use it on a daily basis. First the interviewees were asked to describe the software features they found most helpful in their work. There was much agreement among the users with each software category. Almost all of the word processor users mentioned automatic footnoting and the ability to import graphs as important tools. Other features found useful by word processor users were block structure of paragraphs and print formatting (in WordPerfect) and mail merge and equation editing (in Lotus Manuscript). The Lotus Manuscript user, commenting on that program, said "it is easy to use by a novice and also offers every possible feature an expert could desire.... A novice can perform simple .asks in a half hour." This remark echoes the notion that users go through a two-stage process to learn software, and implies that the degree of utility the user perceives in a program is linked to the degree of expertise he will later achieve.

The AutoCAD users also were fairly consistent in the software features they found important. All of the users listed W-blocks (the ability to split larger drawings into smaller ones), three-dimensional capability, and the AEC template as essential to their work. Also mentioned were the fillet, trim, and extend functions.

An interesting theme that unexpectedly emerged from these interviews was that AutoCAD users did not rely heavily on the program for design. Several said AutoCAD was not intended for design; these users said they relied on board work for the preliminary conceptual sketches that form the basis of design work. They viewed AutoCAD as a drafting tool for producing high quality drawings and reducing repetitive work.

The spreadsheet users had not explored their programs in great depth by the time the interviews were conducted, but mentioned the speeding of numerical calculations, graphing data, and editing data as being important labor-saving features.

On the question of documentation, there was less agreement among users regardless of which packages they used. One WordPerfect user said, "I don't use the documentation. I don't find the documentation to be much use at all." Another said that "the tutorial did a good job, but ... the manual did an average job." The Lotus Manuscript user said that the program's documentation was excellent, but he also supplemented it with information from books and magazine articles.

In general, the researchers noted that as expertise rises there is a limited increase in the use of documentation. Users are most likely to turn to documentation when they have a specific use in mind and believe they can find the information easily in the manual.

The AutoCAD users were also split on their opinions of the documentation. Only one thought the documentation was excellent. One thought it was poor, and another said he didn't use the documentation. It may be significant that the user who felt the AutoCAD documentation was excellent had the most one-on-one training.

Both spreadsheet users said the spreadsheet documentation was well written and useful.

Several respondents said they had bought books designed to supplement the documentation of the programs they used. They reported finding these books valuable because of simplified explanations and organization of information in a sequence relevant to typical users.

Interviewee response was mixed on the subject of getting help from vendors. About half the interviewees never required any assistance from vendors. One user who had needed assistance found the vendor's help to be useful. Two other users found it difficult to get help when they needed it, but this did not seem to affect their perception of the software itself; both gave their work-related package high marks. In general, vendors appear to be the least valued source of supplemental software-related information and help with problems (except those that offer a toll-free customer service line). Two reasons for this lack of reliance are the amount of time it takes to find a person who can help, and the inability of the problem solver to understand the caller's request.

Most of the software users surveyed had, from time to time, used the software in ways differing from what it was actually designed for. Several WordPerfect users, for example, used the search and query functions to search database files or file directories. Some AutoCAD users had employed the software to develop logos, fliers, resumes, and to do homework. One spreadsheet user reported purchasing engineering software written based on a spreadsheet. In general, however, this kind of experimentation was quite limited. It is likely that this lack of experimentation is a result of the lack of a specific need for it, the press of work responsibilities, and the lack of emphasis on developing additional applications.

As previously indicated, almost all the users could identify someone in the organization who could provide assistance on advanced software features. Some users relied on a support network with expertise on particular software features. Likewise, almost all of the users provided assistance to other users in varying degrees, depending on their own level of expertise.

When asked if their work-related software was easy to use, all respondents said it was. One user described the package he used as intuitive. Another who said it was easy to use echoed the theme of the two-stage learning process: one could learn it and be an effective user within 3 days, he said, and in 2 months a user could be fluent.

4   FACTORS AFFECTING THE ORGANIZATION'S ACCEPTANCE OF SOFTWARE

Because the survey reported in the preceding chapter was not quantitative in design or scope, conclusions based on it must be carefully presented. The material that follows supplies a useful perspective for understanding the findings, and is supported by previously completed research on organizational change—particularly the diffusion of innovation. The following topics will be addressed: learning styles that affect how individuals integrate software use into their work routine; the type of support and growth network at USACERL, as outlined by those interviewed; the "need to know" as it influences the growth in expertise; and the impact of software standardization.

Learning Style

Research on adult learning suggests that the ability to use information soon after obtaining it strongly influences motivation to learn.[3] Furthermore, a limited amount of information that meets an immediate, specific need is more useful than a large amount that cannot be applied immediately; small amounts of information presented close to the time of use are considered most valuable by the learner.[4] Also, self-paced learning in situations an adult controls is favored over group settings where competitiveness and fear of error are present.[5]

Those interviewed are typical of adults studied elsewhere in settings as diverse as literacy education, vocational courses, and management development. They wanted to become familiar with each software package in a way that produced increments of "limited mastery," often resulting in a product (e.g., letter, drawing, spreadsheet). That product boosted the user's confidence, which then increased his motivation to continue the mastery process.

The pervasiveness of this style among coworkers at USACERL, combined with the absence of highly regarded training and software documentation, added to its acceptance. Another factor was important among experienced users: they knew what features were most useful from previous experience with other packages; they knew enough to look for similar features in new packages. If basic features were difficult to identify and employ rapidly, user frustration was quick to appear. Some would persist and learn the package because it was a laboratory standard or it seemed to offer the potential of great utility. Others went back to their "tried and true" alternatives and found ways to adapt them to the requirements of the standard.

The "need to know" governed the time and effort the interviewees spent to master a group of easily used features. After an initial period of great activity to achieve initial mastery, most users reached a plateau. Advancement beyond this first plateau occurred through interaction with peers when there was a need to know. Only if a valued coworker mentioned an additional useful feature would the user explore

---

[3] Alan B. Knox, *Adult Development and Learning* (Jossey-Bass, San Francisco, 1977); Roby J. Kidd, *How Adults Learn* (Association Press, New York, 1973); and G.G. Darkenwald and G.A. Larson (Eds.) *New Directions for Continuing Education Reaching Hard-to-Reach Adults*, No. 8 (Jossey-Bass, San Francisco, 1980).

[4] James C. Hall, in Alan B. Knox, *Developing, Administering, and Evaluating Adult Education* (Jossey Bass, San Francisco, 1980).

[5] Alan Tough, *The Adults' Learning Projects* (Learning Concepts, Austin, TX, 1970); Wlodkowski (1985).

and master it. Similar behavior was noted in a study of Army personnel, which indicated that information on a new technology may go unnoticed by personnel who are not facing a problem that the technology can resolve.[6]

Only in limited cases were tutorials used as the method to become familiar with a program. Documentation was often disregarded because it was perceived to be too cumbersome to be of value. Templates and help menus *were* used, however, because they either showed how to perform a function or provided guidance on solving an immediate problem. For technical personnel, training courses also fell into the same category of marginal usefulness. These opinions support the principle that limited information directed to fulfill a specific need provides the user the strongest motivation to learn new skills.

The mode of introduction of software at USACERL has, for the most part, been consistent with the self-paced learning style preferred by the USACERL staff, but this may not be true in the case of support staff. While technical personnel tend to acquire new software skills at their own pace, one interviewee from the support staff said support personnel are often expected to be fully functional shortly after the introduction of a new software package. The existence of a self-paced, unstructured learning method may put too much pressure on support personnel to both learn and be highly productive at the same time. The responses obtai- ,l from this member of the support staff suggest that an early, well planned training program may be highly useful during the introduction of new technologies to support personnel. Because this finding is based on a single interview, no quantitative conclusions can be drawn from it. However, the finding provides insight into an organizational factor that may have a substantial effect on the learning environment for a large segment of USACERL employees. A separate study of this issue would be appropriate.

The majority of those interviewed said they learned about new hardware and software through articles in professional journals and other publications. Several users said they regularly reviewed journals available at USACERL; others indicated that they had personal subscriptions to professional journals or popular microcomputer magazines, such as *PC Magazine*, *Byte*, and *MacWorld*. These sources of information helped the interviewees form opinions about software utility and ease of use. The way this information was integrated with input from peers and coworkers follows closely the two-step information flow process proposed by Rogers. The two-step flow theory states that the mass media bring information to the attention of the public, particularly influential individuals within the social system. Upon learning of the information, individuals will seek out the opinions of others to corroborate the information.[7] This phenomenon has been observed in a study of technology transfer within the Army Corps of Engineers. In that study, a survey of Army engineers revealed that while most of the engineers learn of new technologies through mass media, they prefer to receive information through interpersonal channels.[8]

Support and Growth

Mentors, more experienced coworkers who have an investment in the success of a new colleague,[9] are important in the development of limited and functional expertise among the users interviewed in this study. While they are not exactly textbook examples, mentors at USACERL provide help as part of an

---

[6] Jeffrey J. Walaszek, *The Role of Communications Within Technology Transfer Activities of the U.S. Army Construction Engineering Research Laboratory*, Technical Manuscript (TM) 0-87/1/ADA184334 (USACERL, 1987).

[7] Everett M. Rogers, *The Diffusion of Innovation* (The Free Press, New York, 1962, 3rd Ed., 1983).

[8] Jeffrey A. Walaszek.

[9] Cyril O. Houle, *The Design of Education* (Jossey Bass, San Francisco, 1977), M.S. Knowles, *The Adult Learner* (Gulf Publishing, Houston, TX, 1984).

informal network of coworkers willing and able to assist each other. Within this informal network, those who have appropriate advanced knowledge:

1. Provide 10- to 30-minute introductory tutorials for those just beginning to use a package

2. Answer questions by demonstration, either using documentation or with an explanation based on experience

3. Share with individuals or small groups ideas for making more effective use of the software.

There is no set schedule or pattern for this sharing. User need, interest (to some extent), and the normal pattern of communication about work-related topics influence this type of information sharing activity. In general it appears that these sessions tend to address single items or issues in computing; frequently the topic is how a specific feature could be used to make work more efficient or effective.

The interviews suggest that growth is rapid until a mastery level consistent with the demands of common assignments is reached. Then, need to know and interest in a feature tend to govern the rate and scope of additional learning. When the personnel interviewed were asked to identify features they used regularly, overlap was considerable.

Need to Know

The "need to know" principle combines with the demands of specific assignments to adversely affect the time and interest available for experimentation. Only a few of those interviewed could be characterized as "early adopters," people (as defined by Rogers) who are willing to take the initiative to identify and use advanced software features that could increase productivity, the quality of work output, or the value of data accumulated.[10]

Both individual interest and group support were directed at solving problems associated with completing assignments or meeting regular responsibilities. There was little evidence of the curiosity that encourages experimentation and leads to beneficial new applications of software. Experimentation with software, and the exploitation of advanced features is an area that could improve if USACERL personnel did not budget their learning time so exclusively on the basis of the need to know.

Standardization

The decision to use one software package to promote uniform standards and products is logical from a management perspective. Major benefits of standardization include the establishment of a common format for data exchange and the development of a pool of users of a common program. Some users interviewed found software standardization frustrating. Several experienced users in the sample are determined to employ their own software preferences regardless of the organization's requirements. This apparently has produced little conflict among users or between users and managers; the software standard is enforced with flexibility at USACERL. In fact, the efforts of these users to use their favorite software while complying with the practical necessities of the standards may be considered the only major example of user software experimentation evident at USACERL.

---

[10] Everett M. Rogers.

5 CONCLUSIONS AND RECOMMENDATIONS

Conclusions

This study highlighted two important areas in need of attention through a combination of immediate action and additional study outside the scope of this research.

1. Many of the technical personnel interviewed said their knowledge about and skill in using their work-related software has reached a plateau. They can complete assigned work with their current level of expertise, but analysis of the interviews reveals that they are not motivated to learn additional applications of their software when that additional knowledge and skill cannot be applied directly to specific projects or job responsibilities. Software users generally do not have the time to experiment with the full range of a program's capabilities.

2. Comments by a member of the support staff about the organization's demand for quick mastery of word processing software may signal a problem needing attention throughout the laboratory. Support personnel are often expected to produce a variety of documents on time and at high quality levels. They may not be given enough time to familiarize themselves with new software and may not have access to quality assistance, but they are expected to integrate this technology into their normal work routine without much delay.

Recommendations

In-house organized programs are an effective way to present information in measured amounts. The following recommendations are based on the idea that in-house activities are the best approach to foster both initial and ongoing learning of software.

As a general principle, it is recommended that a fit be established between how personnel learn and how they work. Material presented in some training medium should be related to the needs of the user (employee). Managers must encourage employee growth in computer proficiency, pointing out how enhanced expertise contributes to organizational and individual effectiveness. At USACERL, these duties must fall initially to Division Chiefs, Office Chiefs, and Team Leaders.

The specific recommendations are as follows:

Validate How Software is Mastered

Individual units within USACERL could prepare a brief software guide for new employees that outlines the features of greatest immediate value and the suggested process for mastering the software. The guide should be aimed at both technical and support personnel. The guide should include information on use of tutorials, templates, documentation, and in-house experts who may provide advice. Commonly used terms can be defined. The guide should list application problems regularly encountered and their solutions. The goal of the guide should be to reduce the time a new user needs to reach that first plateau of mastery.

17

Develop a Group of In-House Experts

A few individuals will be interested in achieving a high level of software mastery. Both technical and support personnel could be included in this group of software "pathfinders." They should be given opportunities for appropriate training and experimentation; they should be identified as in-house resources and allowed to help new users and those with specific problems.

Share Features of Demonstrated Value

The informal process of promoting individual software features of proven value should be enhanced with regular opportunities to explain and promote them. This function could best be performed at the division level. If features of potential value to the whole are identified, they should be presented at short sessions or published in the form of one-page guides.

Pace Required Adoption

If a package is designated as the standard for one organization, set a reasonable period of time for personnel to become familiar with the package. Orientation should be included to promote the ease of adoption. Based on the findings of this study, a step-by-step process could be developed to foster efficient implementation of new software technology at a pace reasonable for the user. The objective would be to expose the user to the right information at the right time, and in the right form to promote quick and effective application of the software. One possible process would proceed as follows:

1. Expose potential users to information about the software package through appropriate media. An Army engineer district promoting the use of a commercial program, for example, could distribute a newsletter or reprints of any useful magazine articles describing the software of interest.

2. Survey as many potential users as possible to determine attitudes, questions, and concerns about issues such as ease of use and utility of the proposed software. Based on the survey, identify a set of users with the characteristics of early adopters (i.e, the willingness to explore and use advanced software features).

3. Provide these early adopters with a brief introductory session on the software. Then give them copies of the software and allow them to begin experimenting with it and integrating it into their work routine. Give these early adopters ample time for self-paced learning as well as an in-house resource or toll-free number to answer questions.

4. Interview the early adopters to determine how well the software is being integrated. Identify examples of better functionality or ease of use provided by the software in work-specific settings.

5. Provide copies of the software to the rest of the group. Offer a short introductory session that emphasizes the work-specific applications discovered by the early adopters. Identify the early adopters as the in-house resources to provide one-on-one help.

In general, managers should become actively involved in promoting the growth of software familiarity and expertise among members of the group. No great time investment by the manager would be needed; his or her efforts would be geared to promoting steady, incremental progress by staff members.

Topics for Future Study

This initial study of the ways technical and support personnel at USACERL respond to the introduction of software identified important topics for study that were outside the scope of this basic research. Specific areas to be examined include the following:

1. An in-depth examination of the adoption process for a technical software package, comparing a setting with no formal assistance with one where an adoption process was carefully planned and executed. The literature on education, organizational change, and technology transfer would be used to develop a process to be tested and compared with informal procedures.

2. An analysis of strategies that could be used to overcome resistance to the use of CADD software among architects and engineers. Issues that arose but could not be formally analyzed within the scope of the present study include architect perceptions that CADD software is not an appropriate design tool, and that differences may exist in the way architects and engineers approach CADD software. In addition to gathering more details on the causes of resistance, various methods to improve acceptance would be developed and tested. These could include the use of in-house experts, narrowly defined training programs, and concrete demonstrations of the software's enormous potential.

3. A determination of the special assistance support personnel need to become familiar and suitably proficient with software they must use to complete assigned work.

REFERENCES

Darkenwald, G.G., and Larson, G.A. (Eds.), *New Directions for Continuing Education: Reaching Hard-to-Reach Adults*, No. 8 (Jossey-Bass, San Francisco, 1980).

Etter, Robert T., *Integrating Microcomputers in the Workplace*, unpublished doctoral dissertation (University of Illinois, Urbana, 1989).

Hall, James C, in Alan B. Knox, *Developing, Administering, and Evaluating Adult Education* (Jossey-Bass, San Francisco, 1980).

Houle, Cyril O., *The Design of Education* (Jossey-Bass, San Francisco, 1978).

Kidd, J. Roby, *How Adults Learn* (Association Press, New York, 1973).

Knox, Alan B., *Adult Development and Learning* (Jossey-Bass, San Francisco, 1977).

Knowles, M.S., *The Adult Learner* (Gulf Publishing, Houston, 1984).

Orr, Joel N., "Upwardly Mobile CADD," *PC Magazine*, Vol 6, No. 21 (December 1987).

Rogers, Everett M., *The Diffusion of Innovation*, 3d ed., (The Free Press, New York, 1983).

Stinson, Craig, "Word 5.0 Gains on WordPerfect With a Host of DTP-Like Features," *PC Magazine*, Vol 8, No. 4 (February 1989).

Tough, Alan, *The Adults' Learning Projects* (Learning Concepts, Austin, TX, 1970).

Walaszek, Jeffrey J., *The Role of Communications Within Technology Transfer Activities of the U.S. Army Construction Engineering Research Laboratory*, Technical Manuscript 0-87/1/ADA184334 (USACERL, 1987).

APPENDIX A: SURVEY QUESTIONNAIRE FOR *ORGANIZATIONAL RESPONSE TO THE INTRODUCTION OF NEW COMPUTER SOFTWARE TECHNOLOGY*

BACKGROUND

1. What was your academic major?

2. What advanced training in that or related areas have you received?

3. What is your title and your specific responsibilities?

4. To what extent do you regularly use a computer to complete your work?

5. What software do you rely on most? Why?

6. How did you learn to use this software?

   a. Through a training session
   b. By studying the documentation
   c. With assistance from a coworker
   d. Trial and error

7. In what other ways do you increase your familiarity with computer hardware and software?

   a. By reading journal articles
   b. Through personal experimentation
   c. By attending training sessions
   d. Through conversations with sales or service personnel
   e. Working on projects with coworkers

8. How much support for increasing your expertise do you get from your immediate supervisor?

INTRODUCTION OF SOFTWARE

1. How did you first become familiar with this software package?

2. What features did you feel would be most helpful to you?

3. Did any others you know use the package? What were their impressions?

4. Did you do any research to obtain more information about the package and its features?

5. What, if any, training did you receive before using the package?

6. To what extent, if at all, were you encouraged by your supervisor to use this package? Why?

7. What, if any, initial difficulties did you encounter when you began to use the software?

8. Did you have adequate time to become familiar with the program before you began to use it on a regular basis?

SOFTWARE SATISFACTION AND USAGE

1. What specific features of the software have you found especially helpful in completing your work?

2. Are there any aspects of the documentation that help you in understanding the software features?

3. Have you needed assistance from the software manufacturer? Was it available when needed? Was it of value?

4. As you have become more familiar with the software, have you found other ways to use it?

5. To whom do you turn in the organization for help when a problem occurs in using the software?

6. To what extent have you provided assistance to others using this software for the first time?

7. Is the software package easy for you to use?

APPENDIX B: THE PRODUCTIVITY POTENTIAL OF THREE TYPES OF PC SOFTWARE AT USACERL

Three types of software were chosen as the focus of user interviews: word processors, computer aided drafting and design (CADD), and spreadsheets. Each is important in a distinct category of work performed at USACERL. Some types are generally more important to technical personnel while others are more important to support personnel. Some personnel, however, may use more than one type of software regularly. This appendix outlines how the types of software under consideration here may contribute to greater productivity throughout the laboratory.

Word Processsors

This part of the research was originally to focus on WordPerfect 5.0, which has been designated the in-house standard at USACERL. During interviews, however, it was discovered that more than half of the word processor users were using something other than the lab standard—specifically Microsoft Word 5.0 or, in one case, Lotus Manuscript. For this reason, questions about word processing software were generic rather than brand-specific. This is appropriate since each brand is aimed at a similar niche in the market: the business/office setting, where power and versatility are important because of the volume and variety of documents processed.

Personal computers and word processing software have represented a fundamental revolution in the way business documents and technical manuscripts are typed, edited, and published. Even the simplest word processing packages, when combined with modest microprocessor technology (e.g., the Intel 8086/8088 central processing unit [CPU]), a small, inexpensive hard disk, and a letter-quality printer, can turn a competent individual typist into a virtual typing pool. The combination of software and hardware allows the typist to "capture keystrokes," or create an electronic copy of any document that could be produced on a typewriter—and many documents that could *not*. That file can not only be output through a printer, but can also be stored, retrieved, revised, and reprinted with a fraction of the effort and time the same cycle takes using paper files, a typewriter, and correction fluid. These basic productivity benefits have made their way into so many offices over the past 10 years that they are taken for granted by many. Today, even "low-end" word processors are relatively sophisticated and easy to use; the most powerful commercial packages are readily available and moderately priced.

By its nature, a research lab like USACERL processes a wide variety of documents (e.g., administrative correspondence, technical documentation, newsletters, business forms) with the involvement of personnel varying from entry level typists to senior researchers. Such an organization would seemingly have more to gain than most from a revolution in document processing. In fact, word processing programs such as those used at USACERL offer an almost staggering variety of productivity-enhancing features, but many of these features are not fully exploited by the users (or fully appreciated by the management). Some of the more important productivity aspects of word processing software are presented in the paragraphs that follow.

Text Processing

The ability to display and edit blocks of text on a video monitor represents a full magnitude of productivity improvement over the 80-character displays of memory typewriters, just as those machines were revolutionary when compared with conventional electric typewriters. The word processor's on-screen editing function, when combined with disk storage, is such a powerful document processing tool that many users never feel the need to master the more powerful labor-saving features. To whatever extent that

happens, it represents a considerable amount of untapped productivity. Some intermediate and advanced text-processing capabilities found in the word processing packages used at USACERL include the following:

—On-line dictionary, hyphenator, and thesaurus. A dictionary function can be used either to spellcheck a defined segment of type (e.g., paragraph, page, document, highlighted block) or to look up a specific word. A word processing package comes with a built-in database of hundreds of thousands of words, and generally lets the user create "user dictionaries" of specialized words (technical terms, names) not found in the stock dictionary. The hyphenator automatically breaks words at the end of a line into a correct hyphenization, saving users the trouble of consulting a dictionary. An on-line thesaurus helps a user find the best word to convey a particular meaning by providing a list of alternative synonyms to a selected word.

—Merge. This feature allows the user to produce "editions" of documents, based on a common core, that seamlessly integrate selected information keyed to a number of specific purposes or recipients. The merge function is frequently used to personalize and address administrative correspondence without repetitive typing or the impersonal impact of a form letter. Merge can also be adapted to any category of document that requires a step-by-step filling out of blanks (e.g., project status sheets, evaluations, surveys, training guides, and reports.

—Headers and footers. This function automates the repetitive task of typing standing text at the top or bottom of each page in a multipage document (e.g., chapter titles, page numbers).

—Automatic footnoting and endnoting. These features enable the user to enter an alphanumeric character or typographical symbol at any specific point in text and link it to a small subfile containing documentation for a fact or a textual footnote. When a new note is added between existing ones, all those following the new note automatically renumber themselves while remaining linked to their text. Footnote text is automatically printed on the same page that its note appears regardless of how the document has been edited since the note was first entered. Automatic footnoting in particular can remove most of the difficulty from one of the more problematic aspects of typing technical manuscripts.

—Table and equation editors. These facilitate the arrangement of text and/or numbers on a page to help simplify and clarify printed technical information. A table editor can make it relatively easy to arrange data into orderly columns and rows. An equation editor allows the user to access mathematical and scientific symbols and arrange them in the proper spatial interrelationships to express complex calculations and formulae with visual clarity. Multiple equations in a document are automatically numbered, and the numbering function is self-editing for ease of adding, deleting, or moving equations.

Typeface control

When combined with a printer, word processors can put a variety of different typefaces and styles onto a page. The power and quality of this feature largely depends on the printer being used, but a program that makes it easy to change or manipulate available typefaces gives the typist (or author) the potential of achieving a legibility and professional look that was previously only available at considerable cost through typesetting vendors. This is more than a mere cosmetic enhancement: by outputting a word processing file through a laser printer, the user can easily create presentation-quality documents without the costs in money, time, and duplicated effort associated with using a typesetting contractor. For widely circulated work, the electronic word processing file can be output through a vendor's compatible phototypesetter without the necessity of rekeying the document; this simple benefit alone makes an entire cycle of keyboarding, proofreading, and revising obsolete in published documents.

Word processors make it easy for the user to employ typographic refinements such as italics, boldface, superscript/subscript characters, and mathematical and scientific symbols. These can help organize a document, direct the reader's attention to important points, and express technical ideas in ways a typewriter never adequately could.

A word processor with any degree of sophistication includes the ability to create "stylesheets," sets of typographical specifications (e.g., font, weight, size) linked to a recurring functional category of type (e.g., section title, figure caption, body type). In some packages, styles may include functions like margins, hanging indents, and paragraph spacing. After the stylesheets are initially input, the user can rapidly format a document simply by highlighting text with the cursor, then hitting a short combination of keystrokes that executes all specifications for the desired style.

Page Layout Capabilities

During the last half of the 1980s the term desktop publishing (DTP), or electronic publishing, was coined to describe a powerful subcategory of word processing software that allows the user to duplicate the work of a publication designer, typesetter, layout artist, and even a graphic artist. DTP software generally has not included the most powerful automated text processing capabilities of the high-end word processing programs, but far surpasses the word processor's ability to display and integrate graphics. DTP software, supplemented by a "mouse" input device, enables the user to quickly and easily make major changes in a document's appearance or layout. Frequently all this requires is pointing at an object with the cursor (a graphic, for example), clicking a mouse button, and dragging the object to the desired new location on the page.

In recent years the distinction between word processors and DTP software has been disappearing. The most powerful DTP applications are starting to offer more sophisticated text processing features while word processing packages now include a number of functions previously found only in DTP applications. WordPerfect 5.0, USACERL's standard in-house word processing program, permits the user to lay out text in newspaper-style columns, import graphics from external files and move them on the page, and preview a document's actual appearance on the video display before outputting it through the printer. Such features have begun to revolutionize the production of technology transfer documents and newsletters in some publishing-oriented offices at USACERL, but the potential productivity and quality enhancements offered by powerful word processors (and their DTP-like features) are only starting to be understood in the general user community.

Macros

Macros—small routines or subprograms that automate a wide variety of miscellaneous keyboarding tasks—are not exclusive to word processing packages. They can be used to simplify the coding of specialized typographic applications (e.g., superscripts and subscripts) or create customized procedures for document backup and filing. WordPerfect comes with a powerful macro function whose potential as a labor-saving tool is limited mainly by the user's imagination and opportunity to develop applications.

Computer Aided Drafting and Design

This category of productivity-enhancing software is used in a number of technical areas throughout USACERL, especially by architects, criteria designers, and all engineering disciplines. CADD packages vary widely in power and capability, from modest drawing programs costing a few hundred dollars to full-featured, three-dimensional applications costing thousands of dollars and requiring the most sophisticated microcomputer hardware on the market. One two-dimensional program, however—AutoCAD—is more

25

widely used than any other at USACERL because of its rich set of user tools and relatively low purchase price. Therefore, any discussion of CADD in this report relates specifically to AutoCAD unless otherwise specified.

Much technical work at USACERL involves drafting at some point. Unless nothing more than a rudimentary sketch is required, the drafting process is frequently a painstaking and time-consuming chore. Furthermore, designs that incorporate many repeated elements (e.g., furniture layout, engineering symbology) can be tedious, and tedium invites mistakes on the part of the person drafting. Considering the overall effort required to render competent, error-free technical drawings, CADD is a tool with tremendous potential to boost the productivity of anybody who spends much time drafting.

Labor-Saving Aspects

The basic value of a CADD package is to take the effort out of the more basic chores of drafting as well as the more intricate ones. A CADD program makes it easy to create, position, and manipulate lines, curves, and polygons (and, in the case of three-dimensional programs, polyhedrons). Even for the execution of simple drawings, the productivity-enhancement capabilities of CADD software should not be underestimated: CADD eliminates the need for corrections made necessary by a slip of the hand or the "temperamental" nib of a technical pen. In fact, all essential tools for drafting—pen, straightedge, T-Square, triangle, compass, templates—are integrated into good CADD programs such as AutoCAD.

AutoCAD's predefined commands for geometric construction include automatic polygon closing and commands to produce absolute, relative, and polar coordinates. More complex geometric construction features (e.g., cubic splines, conic splines, bezier curves) can be created by the AutoCAD user with AutoLISP, a programming language included in the package. (Some CADD packages come with these features and others as part of the standard tool set.) A very useful tool is the undo/redo command, which makes it very easy to correct errors or compare two variations of an item in quick sequence.

One of the more important productivity enhancements offered by CADD software is the capability to edit a preexisting drawing with much more ease than possible with pen and ink. Drawn objects can be trimmed and moved; objects can be grouped and manipulated as a single item; objects (and groups of objects) can be rescaled or stretched with little effort. An extremely useful tool is the "snap" function, which makes it possible to execute precision alignment among lines, arcs, and objects without the demands on eye-hand coordination posed by conventional drafting techniques. The snap function makes it easy to create perpendiculars to an exact point on a line, for example, or tangents to an arc, ellipse, or circle.

CADD programs also save labor required for various dimensioning activities. AutoCAD's standard dimensioning commands control baseline, angular, and continuing dimension functions. Several methods are included for automatically calculating areas and perimeters. More elaborate programs include the ability to calculate center of gravity and other mass properties; the AutoCAD user may create such capabilities in the software through programming with AutoLISP.

Iteration and Exploration of Design Alternatives

The very nature of CADD files (i.e., digital information easily retrievable from magnetic storage media) makes it easy to create and recreate hard copies of a design at various stages. With the proper hardware and software configurations, elements from one drawing can be incorporated into another with little effort. In AutoCAD, any drawing can be transformed into a symbol that can easily be placed into another drawing. Furthermore, repeating elements can be introduced into a large project quickly and easily. If several floors in a building are based on the same design, for example, an architect can easily and quickly make multiple copies of that design and add them to the overall drawing.

Perhaps the greatest potential for user productivity and creativity resides in the ease with which alterations (major and minor) can be made to a basic design. This capability offers a potential benefit that is easy to understand if somewhat difficult to quantify: an individual user can employ CADD software to explore many more design possibilities than would be possible using conventional pen-and-ink methods in the same available time. The more design possibilities explored, the more likely it is that a user can create designs resulting in higher product quality and lower life-cycle cost. Productivity benefits of that kind would be of great interest in any research and development setting.

Spreadsheet Software

Spreadsheet packages are applications used to organize interrelated data (mainly numbers) into two-dimensional arrays of "cells" (units) that allow the user to extract meaning from an otherwise cumbersome collection of numbers. A checkbook ledger is an example of the kind of information that can be managed by a spreadsheet, but the major commercial programs available today (e.g., Lotus 1-2-3, Microsoft Excel) are powerful tools widely used throughout the business, engineering, and science fields.

The basic spreadsheet application is the table, a series of rows and columns into which data are entered, stored, and manipulated. Rows and columns are composed of cells, each of which contains a discrete packet of information. A cell is not limited to recording a numerical value, but may contain a variable or a mathematical formula that is automatically applied to other specified cells. A cell may hold a formula that, for example, automatically adds line item values for a specified budgetary account. Whenever the value of a line item cell is updated, the formula updates the total in the cell governed by the formula.

The basic tools for manipulating data are functions that are familiar from word processing applications: data (including formulas and data subsets) can be entered, deleted, copied, and moved. Documents can be saved to magnetic storage media, retrieved, updated (edited), and printed. Cell format can be specified for a variety of parameters, including width and depth, typeface style, size, and weight, and alignment of data. Rules (vertical and horizontal lines) can be added (or deleted) for visual clarity.

With the proper configuration of hardware and software, a number of advanced data management capabilities are available. 'Live' links can be created among spreadsheet documents, so changes made in one document can alter specified data in another. Different parts of a large document, and even different documents, may be displayed in separate (simultaneously viewable) windows. Data (or blocks of data) may also be moved from one window to another. Additionally, spreadsheets may be exported to documents created by a dissimilar application, such as a word processor. When the user employs a spreadsheet and word processor published by the same company (e.g., Lotus 1-2-3 and Lotus Manuscript), transfer of data between applications may be transparent and effortless, avoiding the translation problems that frequently occur when converting information from one filing format to another.

Spreadsheets have been a boon to budgeteers and managers over the past decade, but the better packages now incorporate graphics capabilities that make them much more than an accountant's tool. A spreadsheet's graphics functions can represent complex files of numerical data as visual images that provide immediate insight and clarity. For example, the software may convert raw numbers into a variety of charts or graphs automatically, with no computation (and little other input) by the user. Formats may include the line graph, bar graph, combination graph, scatter graph, pie chart, and x-y (cartesian) graph. Some packages (e.g., Wingz) invest much code in sophisticated graphics modules that include three-dimensional (three-axis) graphing and extensive control of color rendition on compatible output devices. These graphics tools offer the PC user a powerful set of presentation tools which, only a few years ago, were available only through the services of graphic artists.

Although spreadsheet software was not originally intended to be a focus of this research, interviews with USACERL staff indicated that this technology offered the organization important enough potential benefits to warrant inclusion in the study.

USACERL DISTRIBUTION

Chief of Engineers
ATTN: CEHEC-IM-LH (2)
ATTN: CEHEC-IM-LP (2)
ATTN: CEMP-EC
ATTN: CERD-L
ATTN: DAEN-ZCZ
ATTN: DAEN-ZCI
ATTN: DAEN-ZCM
ATTN: DAEN-ZCE

US Army Engineer Districts
ATTN: Library (41)

US Engineer Divisions
ATTN: Library (14)

Fort Belvoir, VA 22060
ATTN: CECC-R

FORSCOM
FORSCOM Engr, ATTN: Spt Det. 15071

AMC - Dir., Inst., & Svcs.
ATTN: DEH

HSC
Walter Reed AMC 20307
ATTN: Facilities Engineer
Ft. Sam Houston AMC 78234
ATTN: HSLO-F
Fitzsimons AMC 80045
ATTN: HSHG-DEH

INSCOM - Ch, Instl. Div.
Vint Hill Farms Station 22186
ATTN: IAV-DEH
Arlington Hall Station 22212
ATTN: Engr & Hsg Div

USA AMCCOM 61299
ATTN: Library

TRADOC
HQ, TRADOC, ATTN: ATEN-DEH 23651
ATTN: DEH

WESTCOM
Fort Shafter
ATTN: DEH
ATTN: APEN-A

CECRL, ATTN: Library 03755

CEWES, ATTN: Library 39180

NAVFAC, ATTN: Division Offices (11)

US Army Env. Hygiene Agency
ATTN: HSHB-ME 21010

National Guard Bureau 20310
Installation Division

American Public Works Association 60637

Nat'l Institute of Standards & Tech 20899

Defense Technical Info. Center 22304
ATTN: DTIC-FAB (2)

07/91
97